REMARKS

Claims 8, 9, 11, 27, 39 and 42-50 have been amended. Claim 24 has been canceled claims. Accordingly, claims 8, 9, 11, 21, 25 – 30, 32 – 37, 39, 40 and 42 – 50 are pending. Reconsideration is respectfully requested in light of the following remarks.

Section 112, Second Paragraph, Rejection:

The Office Action rejected claims 11 and 42-50 under 35 U.S.C. § 112, second paragraph, as being indefinite. The claims have been amended for clarification. Applicants submit that these rejections have now been overcome.

Section 103(a) Rejections:

The Office Action rejected claims 25, 26, 28-30, 36, 37, 42-44 and 46-48 under 35 U.S.C. § 103(a) as being unpatentable over OMG (Interface Repository OMG TC Document 94-11-7) (hereinafter "OMG") in view of Nelson et al (US Pat. 5,577,252) (hereinafter "Nelson") and SOM (SOMobjects Developer Toolkit Users Guide, version 2.0, pages 7-1 to 7-16) (hereinafter "SOM"). Applicants respectfully traverse this rejection for at least the following reasons.

With respect to claim 25, contrary to the Examiner's assertion, the cited art does not teach or suggest an interface repository including a repository naming context, and a prefix naming context subordinate to the repository naming context, the prefix naming context serving as a root naming context for at least one interface definition language declaration. In prior art interface repositories, such as described in the OMG reference cited by the Examiner, *the repository naming context serves as the root naming context for all objects as defined by their corresponding interface definition language (IDL) declaration.* This is required so that an object corresponding to a particular IDL declaration can be uniquely resolved within the interface repository.

If two different versions of the same object type are desired, then either separate repositories must be used or different object IDs must be used. These two different approaches are illustrated in Fig. 1 at p. 11 of the Examiner's OMG reference. To support different versions of the interface object "Doc", Fig. 1 of OMG illustrates that two different repositories may be used. For example, in the "SoftCo, Inc." repository interface object Doc is declared under module softco (i.e. at softco/Doc). A different version of Doc is declared in a separate repository named "Customer, Inc." under the softco module which is under the testfirst module (i.e. at testfirst/softco/Doc). Fig. 1 of OMG also illustrated that different versions of an object may be distinguished by declaring the objects with different IDs (Doc id 123 and Doc id 456). These two different techniques are described in OMG on p. 11 and p. 13. *In the OMG reference, if a defined object is loaded to a particular repository naming context, then that repository naming context is the root naming context.* In contrast, claim 25 recites a prefix naming context subordinate to the repository naming context, the prefix naming context serving as a root naming context for at least one interface definition language declaration. As recited in claim 25, the prefix naming context serves as the root naming context to the IDL declaration instead of the repository naming context. This allows two objects that would otherwise have the same naming context to be resolved separately.

In this aspect, the naming service of Nelson is no different than the naming service of OMG. In Fig. 5 of Nelson, Name Server A provides a root context for all objects. Nelson – col. 6, lines 45-53. The Examiner refers to naming context "C" in "C/J" in Fig. 5 of Nelson. However, "C" is an intermediate naming context, not the root naming context for object "J". In Fig. 5, object "J" is declared at D/C/J and loaded in the root repository context of Name Server A. Nelson – col. 6, lines 45-53. In Nelson, if another version of the same object declaration D/C/J was loaded in Name Server A, it would conflict with the existing D/C/J object. The interface repository in the SOM reference works just like the repositories in OMG and Nelson in this aspect. In contrast, claim 25 recites a prefix naming context subordinate to the repository naming context that would allow two versions of an object declared at D/C/J, for example, to be resolved separately by creating one or more prefix naming contexts. In all three of the cited

references, an object declared at D/C/J and loaded in a repository would have the repository naming context as its root naming context such that only one version of the D/C/J object could be loaded and uniquely resolved. Clearly none of the cited references teach or suggest an interface repository including a repository naming context, and a prefix naming context subordinate to the repository naming context, the prefix naming context serving as a root naming context for at least one interface definition language declaration. Similar arguments apply in regard to independent claims 36 and 42.

Further in regard to claim 25, contrary to the Examiner's assertion, the cited art does not teach or suggest an interface repository loader that accepts as input parameters a specified interface definition language file containing at least one interface definition language declaration, and a specified prefix name, and installs the at least one interface definition language declaration in a prefix naming context having the prefix naming context in the interface repository. The Examiner refers to section 7.2 of the SOM reference. However, this section of SOM teaches an Interface Repository emitter to "create an Interface Repository named 'newcls.ir'." (emphasis added). The portion of SOM cited by the Examiner teaches creating an interface repository named newcls.ir located at c:\myfiles\newcls.ir. Thus, c:\myfiles\ specifies the location for the repository itself, not a prefix naming context within the interface repository. SOM does not teach or suggest any prefix naming context within and subordinate to the newcls.ir repository. A similar argument applies in regard to independent claim 42.

Allowable Subject Matter:

Claim 11 was objected to, but otherwise allowable if rewritten or amended to overcome the rejection under 35 U.S.C. § 112, second paragraph. Claims 27 and 39 were objected to as being dependent upon rejected base claims, but otherwise allowable if rewritten in independent form. Claims 45, 49 and 50 were objected to, but otherwise allowable if rewritten or amended to overcome the rejection under 35 U.S.C. § 112, second paragraph and if rewritten in independent form. Claims 11, 27, 39, 45, 49 and 50 have been amended as described above. Claims 8, 9 and 11 were also amended to delete

text inadvertently added in the previous amendment. As such, Applicants assert that claims 11, 27, 39, 45, 49 and 50 are in condition for allowance.

Allowed Claims:

Claims 8, 9, 21, 32-35 and 40 are allowed.

CONCLUSION

Applicants submit the application is in condition for allowance, and notice to that effect is respectfully requested.

If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5681-82301/RCK.

Also enclosed herewith are the following items:

☒ Return Receipt Postcard

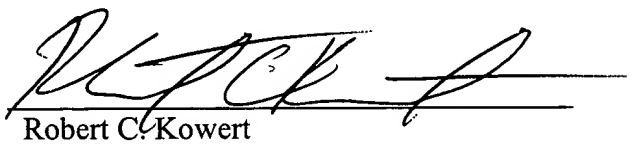
☐ Petition for Extension of Time

☐ Notice of Change of Address

☐ Fee Authorization Form authorizing a deposit account debit in the amount of $ for fees ( ).

☐ Other:

Respectfully submitted,



Robert C. Kowert
Reg. No. 39,255
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8850

Date: August 11, 2004

APPENDIX A

82301		5/24/1999	09/317,714	Method and System for Type Identification for Multiple Object Interfaces in a Distributed Object Environment	Kessler, Lim Vanderbilt, Powell, Chen, Hare, Snyder
82400	7914 / 0931	03/04/96	08/610,004	Multibyte Locale for Single Byte Languages	Watanabe & Shannon
82401	7914 / 0931	1/10/2001	09/758,762	Multibyte Locale for Single Byte Languages	Watanabe & Shannon
84500	Assignment attached	7/7/1995	08/499,485	System and Method to Transparently Integrate Private Key Operations From a Smart Card with Host-Based Encryption Services	Samar
84800	Assignment attached	7/5/1995	08/498,366	Use of Interpositioning to Provide Integrity and Compatibility Monitor of Computer Interfaces	Hildenbrand
85700	011270 / 0554	10/31/2000	09/702,524	Method and Apparatus for Providing Computer-Based Help	Robert A. Yennaco
86402	8768 / 0155	8/14/1997	08/911,091	Method, Apparatus and Computer Program Product for Using Deferred Execution for a Tiling Pull Model in a Tiled Image Processing Architecture	Furlani, Ohlson, & Inman